Elasticsearch Metadata Extractor plugin

Elasticsearch metadata extractor plugin is used to extract metadata from file (local or from server) and then index them into chosen index.

- Easy to use with single endpoint
- Using powerful and stable Apache libraries for extraction
- Written in JAVA

Installation

- Download **metadata-extractor-x.y.z.zip** (x.y.z represents version of elasticsearch, version used in this example: **7.5.0**) from repository

```
$ wget "https://github.com/opendatalabcz/document-metadata/raw/master
```

- Download and extract elasticsearch with the same version as metadata-extractor plugin

```
$ wget "https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.5.0
$ tar -xvf ./elasticsearch-7.5.0-linux-x86_64.tar.gz
```

- Install metadata-extractor plugin (answer **y** to plugin permission)

```
$ ./elasticsearch-7.5.0/bin/elasticsearch-plugin install file://$PWD/metadata-ex
```

- Start elasticsearch with installed metadata-extractor plugin

```
$ ./elasticsearch-7.5.0/bin/elasticsearch
```

TIPS:

- Always keep **same version** of plugin (zip file) and elasticsearch
- You can check installed plugin description with command:

```
$ ./elasticsearch-7.5.0/bin/elasticsearch-plugin list --verbose
```

- You can remove installed plugin with command:

```
$ ./elasticsearch-7.5.0/bin/elasticsearch-plugin remove metadata-extractor
```

- If you are installing plugin on **Windows**, path for file looks like this:

```
./elasticsearch-plugin install file:\\\C:\metadata-extractor-7.5.0.zip
```

Tutorial

Request:

```
PUT /_extract_metadata
POST /_extract_metadata
```

Request body

`index` (required) (String)

- specify the output index

`path` (required) (String)

- url path to the file from which you want to extract metadata
- local (file://{path_to_file}) or from server (https://{path_to_file})

`_id` (optional) (String)

- elasticsearch use it as document id

`extras` (optional) (JSON object)

- this object will be saved beside metada object in elasticsearch document
- JSON structure object

Example 1

Simple request extracting metadata from local pdf file on linux and indexing it to specified index in elasticsearch.

request

```
curl -X PUT "http://localhost:9200/_extract_metadata" -H 'Content-Type: applicat
{
        "index":"test",
        "path":"file:///home/tester/doc1.pdf"
}'
```

es document


```
{
  "_index": "test",
  "_type": "_doc",
  "_id": "_CSBMXEBbV9ku85xj6_w",
  "_version": 1,
  "_score": 0,
  "_source": {
    "metadata": {
      "document_metadata_dict": {
        "CreationDate": "D:20070223175637+02'00'",
        "Producer": "OpenOffice.org 2.1",
        "Author": "Evangelos Vlachogiannis",
```

```
        "Creator": "Writer"
      },
      "document_metadata_xml": {},
      "pages_metadata": []
    }
  }
}
```

Example 2

Complex request extracting metadata from online pdf source, with also specified document **_id** and **extras** data.

request

```
curl -X PUT "http://localhost:9200/_extract_metadata" -H 'Content-Type: applicat
{
        "index":"test",
        "_id":"test_2",
        "path":"https://file-examples.com/wp-content/uploads/2017/10/file-sample
        "extras":{
                "test_obj1":{
                        "type_1":"test_type_1",
                        "type_2":"test_type_2"
                }
        }
}'
```

es document

```
{
  "_index": "test",
  "_type": "_doc",
  "_id": "test_2",
  "_version": 1,
  "_score": 0,
  "_source": {
    "metadata": {
      "document_metadata_dict": {
```

```
        "CreationDate": "D:20170816144413+02'00'",
        "Producer": "LibreOffice 4.2",
        "Creator": "Writer"
      },
      "document_metadata_xml": {},
      "pages_metadata": []
    },
    "extras": {
      "test_obj1": {
        "type_2": "test_type_2",
        "type_1": "test_type_1"
      }
    }
  }
}
```


Versions

All available versions are in releases package

- each zip file contains plugin descriptor, policy file and jar files
- plugin will be correctly installed and run on elasticsearch version same as plugin version (e.g. metadata-extractor-7.5.0.zip will run correctly on elasticsearch version 7.5.0 -> last 3 digits with dots are representing the version.)

Development

Steps for adding new extractor class:

- **create class** in: implementation package which implements abstract extraction module
- **extractMetadata** function is resposible for extracting metadata from given file and returning them as JSON Object

- **getSupportedExtentions** function is responsible for returning array of strings (representing supported extentions, e.g. { “doc” , “docx” })

Documentation: javadoc